JC20 Rec'd PCT/PTO SEP 26 2001

| FORM-PTO-1390 (Rev. 12-29-99) U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE **TRANSMITTAL LETTER TO THE UNITED STATES DESIGNATED/ELECTED OFFICE (DO/EO/US) CONCERNING A FILING UNDER 35 U.S.C. 371** | | ATTORNEY'S DOCKET NUMBER 032326-169<br>U S APPLICATION NO (If known, see 37 C F.R 1 5) Unassigned 09/937396 |
|---|---|---|
| INTERNATIONAL APPLICATION NO. PCT/FR00/00603 | INTERNATIONAL FILING DATE 13 March 2000 | PRIORITY DATE CLAIMED 26 March 1999 |
| TITLE OF INVENTION COUNTERMEASURE PROCEDURES IN AN ELECTRONIC COMPONENT IMPLEMENTING AN ELLIPTICAL CURVE TYPE PUBLIC KEY ENCRYPTION ALGORITHM | | |
| APPLICANT(S) FOR DO/EO/US Jean-Sébastien CORON | | |

Applicant herewith submits to the United States Designated/Elected Office (DO/EO/US) the following items and other information:

1. ☒ This is a **FIRST** submission of items concerning a filing under 35 U.S.C. 371.
2. ☐ This is a **SECOND** or **SUBSEQUENT** submission of items concerning a filing under 35 U.S.C. 371.
3. ☒ This is an express request to begin national examination procedures (35 U.S.C. 371(f)) at any time rather than delay examination until the expiration of the applicable time limit set in 35 U.S.C. 371(b) and the PCT Articles 22 and 39(1).
4. ☒ A proper Demand for International Preliminary Examination was made by the 19th month from the earliest claimed priority date.
5. ☒ A copy of the International Application as filed (35 U.S.C. 371(c)(2))
   - a. ☐ is transmitted herewith (required only if not transmitted by the International Bureau).
   - b. ☒ has been transmitted by the International Bureau.
   - c. ☐ is not required, as the application was filed in the United States Receiving Office (RO/US)
6. ☒ A translation of the International Application into English (35 U.S.C. 371(c)(2)).
7. ☒ Amendments to the claims of the International Application under PCT Article 19 (35 U.S.C. 371(c)(3))
   - a. ☐ are transmitted herewith (required only if not transmitted by the International Bureau).
   - b. ☐ have been transmitted by the International Bureau.
   - c. ☐ have not been made; however, the time limit for making such amendments has NOT expired.
   - d. ☒ have not been made and will not be made.
8. ☐ A translation of the amendments to the claims under PCT Article 19 (35 U.S.C. 371(c)(3)).
9. ☐ An oath or declaration of the inventor(s) (35 U.S.C. 371(c)(4)).
10. ☒ A translation of the annexes to the International Preliminary Examination Report under PCT Article 36 (35 U.S.C. 371(c)(5)).

**Items 11. to 16. below concern other document(s) or information included:**

11. ☒ An Information Disclosure Statement under 37 CFR 1.97 and 1.98.
12. ☐ An assignment document for recording. A separate cover sheet in compliance with 37 CFR 3.28 and 3.31 is included.
13. ☒ A FIRST preliminary amendment.
    ☐ A SECOND or SUBSEQUENT preliminary amendment.
14. ☐ A substitute specification.
15. ☐ A change of power of attorney and/or address letter.
16. ☐ Other items or information:

JC16 Rec'd PCT/PTO SEP 26 2001

| U.S. APPLICATION NO. (If known, see 37 C.F.R. 1.50) | INTERNATIONAL APPLICATION NO | ATTORNEY'S DOCKET NUMBER |
|---|---|---|
| Unassigned 09/937396 | PCT/FR00/00603 | 032326-169 |

| | CALCULATIONS | PTO USE ONLY |
|---|---|---|
| 17. ☒ The following fees are submitted: | | |
| **Basic National Fee (37 CFR 1.492(a)(1)-(5)):**<br>Neither international preliminary examination fee (37 CFR 1.482) nor international search fee (37 CFR 1.445(a)(2)) paid to USPTO and International Search Report not prepared by the EPO or JPO .......... $1,000.00 (960)<br>International preliminary examination fee (37 CFR 1.482) not paid to USPTO but International Search Report prepared by the EPO or JPO .......... $860.00 (970)<br>International preliminary examination fee (37 CFR 1.482) not paid to USPTO but international search fee (37 CFR 1.445(a)(2)) paid to USPTO ............ $710.00 (958)<br>International preliminary examination fee paid to USPTO (37 CFR 1.482) but all claims did not satisfy provisions of PCT Article 33(1)-(4) ............. $690.00 (956)<br>International preliminary examination fee paid to USPTO (37 CFR 1.482) and all claims satisfied provisions of PCT Article 33(1)-(4) ................. $100.00 (962) | | |
| **ENTER APPROPRIATE BASIC FEE AMOUNT =** | $ 860.00 | |
| Surcharge of **$130.00 (154)** for furnishing the oath or declaration later than ☐ 20 ☐ 30 months from the earliest claimed priority date (37 CFR 1.492(e)). | $ -0- | |

| Claims | Number Filed | Number Extra | Rate | | |
|---|---|---|---|---|---|
| Total Claims | 13 -20 = | -0- | X$18.00 (966) | $ -0- | |
| Independent Claims | 1 -3 = | -0- | X$80.00 (964) | $ -0- | |
| Multiple dependent claim(s) (if applicable) | | | + $270.00 (968) | $ -0- | |

| | CALCULATIONS | PTO USE ONLY |
|---|---|---|
| **TOTAL OF ABOVE CALCULATIONS =** | $ 860.00 | |
| Reduction for 1/2 for filing by small entity, if applicable (see below). | $ -0- | - |
| **SUBTOTAL =** | $ 860.00 | |
| Processing fee of **$130.00 (156)** for furnishing the English translation later than ☐ 20 ☐ 30 months from the earliest claimed priority date (37 CFR 1.492(f)). + | $ -0- | |
| **TOTAL NATIONAL FEE =** | $ -0- | |
| Fee for recording the enclosed assignment (37 CFR 1.21(h)). The assignment must be accompanied by an appropriate cover sheet (37 CFR 3.28, 3.31). **$40.00 (581)** per property + | $ -0- | |
| **TOTAL FEES ENCLOSED =** | $ 860.00 | |
| | **Amount to be: refunded** | $ |
| | **charged** | $ |

a. ☐ Small entity status is hereby claimed.

b. ☒ A check in the amount of $ 860.00 to cover the above fees is enclosed.

c. ☐ Please charge my Deposit Account No. 02-4800 in the amount of $______ to cover the above fees. A duplicate copy of this sheet is enclosed.

d. ☒ The Commissioner is hereby authorized to charge any additional fees which may be required, or credit any overpayment to Deposit Account No. 02-4800. A duplicate copy of this sheet is enclosed.

**NOTE: Where an appropriate time limit under 37 CFR 1.494 or 1.495 has not been met, a petition to revive (37 CFR 1.137(a) or (b)) must be filed and granted to restore the application to pending status.**

SEND ALL CORRESPONDENCE TO:

James A. LaBarre
BURNS, DOANE, SWECKER & MATHIS, L.L.P.
P.O. Box 1404
Alexandria, Virginia 22313-1404
(703) 836-6620

[signature]
SIGNATURE

James A. LaBarre
NAME

28,632
REGISTRATION NUMBER

09/937396
JC16 Rec'd PCT/PTO SEP 26 2001

Patent
Attorney's Docket No. 032326-169

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

| | | |
|---|---|---|
| In re Patent Application of | ) | |
| | ) | |
| Jean-Sébastien CORON | ) | Group Art Unit: Unassigned |
| | ) | |
| Application No.: Unassigned | ) | Examiner: Unassigned |
| | ) | |
| Filed: September 26, 2001 | ) | |
| | ) | |
| For: COUNTERMEASURE PROCEDURES IN AN ELECTRONIC COMPONENT IMPLEMENTING AN ELLIPTICAL CURVE TYPE PUBLIC KEY ENCRYPTION ALGORITHM | ) | |



## PRELIMINARY AMENDMENT

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

Prior to examination and the calculation of filing fees, kindly amend the above-identified application as follows:

## IN THE SPECIFICATION:

Page 1, immediately following the title appearing on lines 1-3, insert the following:

--This disclosure is based upon French Application No. 99/03921, filed on March 26, 1999 and International Application No. PCT/FR00/00603, filed March 13, 2000, which was published on October 5, 2000 in a language other than English, the contents of which are incorporated herein by reference.

**Background of the Invention**--

Page 11, between lines 21 and 22, insert the following heading:

--**Description of the Invention**--

Add the following Abstract:

--Elliptical curve based cryptographic algorithms are public key algorithms offering a shorter calculation time and smaller key sizes in comparison with RSA. In a smart card type environment, these algorithms are vulnerable to differential power analysis (DPA) attacks. The disclosed invention provides a countermeasure procedure enabling positive action to be taken against DPA-type attacks. The countermeasure does not reduce performance and is easy to use in a smartcard type component.--



**IN THE CLAIMS:**

Kindly replace claims 1-13, as follows.

1. (Amended) A countermeasure method in an electronic component implementing an elliptical curve type public key encryption algorithm, wherein a point P on the elliptical curve is represented by the projective coordinates (X, Y, Z) such that x=X/Z and y=Y/Z^3, x and y being the coordinates of the point on the elliptical curve in terms of affine coordinates, said curve comprising n elements and being defined on a finite field GF(p), where p is a prime number and the curve has the equation y^2=x^3+a*x+b, or defined on a finite field GF(2^n), with the curve having the equation y^2+x*y=x^3+a*x^2+b, where a and b are integer parameters, the method comprising the steps of:

1) Drawing at random an integer l such that $0<l<p$;

2) For a point P represented by projective coordinates (X1, Y1, Z1), calculating $X'1=l^2*X1$, $Y'1=l^3*Y1$ and $Z'1=l*Z1$, to define the coordinates of the point $P'=(X'1,Y'1,Z'1)$; and

3) Calculating an output point $Q=2*P'$ that is represented by projective coordinates (X2, Y2, Z2).

2. (Amended) A countermeasure method according to Claim 1, wherein the elliptical curve is defined on the finite field GF(p), and the step of calculating Q includes the following steps:

Calculate $M=3*X'1^2+a*Z'1^4$;

Calculate $Z2=2*Y'1*Z'1$;

Calculate $S=4*X'1*Y'1^2$;

Calculate $X2=M^2-2*S$;

Calculate $T=8*Y'1^4$; and

Calculate $Y2=M*(S-X2)-T$.

3. (Amended) A countermeasure method according to Claim 1, wherein the elliptical curve is defined on the finite field GF(p), and further including the following steps:

Drawing at random a non-zero integer l of $GF(2^n)$;

Replacing X0 with $l^2*X0$, Y0 with $l^3*Y0$ and Z0 with $l*Z0$;

Drawing at random a non-zero integer m of GF(2^n);

Replacing X1 with m^2*X1, Y1 with m^3*Y1 and Z1 with m*Z1; and

Calculating R=P+Q.

4. (Amended) A countermeasure method according to Claim 1, further including the calculation of the projective coordinates of the point R=(X2,Y2,Z2) such that R=P+Q with P=(X0,Y0,Z0) and Q=(X1,Y1,Z1) according to the following steps, with the calculations in each of the steps being effected modulo p:

Replacing X0 with l^2*X0, Y0 with l^3*Y0 and Z0 with l*Z0;

Drawing at random an integer m such that 0<m<p;

Replacing X1 with m^2*X1, Y1 with m^3*Y1 and Z1 with m*Z1;

Calculate U0=X0*Z1^2;

Calculate S0=Y0*Z1^3;

Calculate U1=X1*Z0^2;

Calculate S1=Y1*Z0^3;

Calculate W=U0-U1;

Calculate R=S0-S1;

Calculate T=U0+U1;

Calculate M=S0+S1;

Calculate Z2=ZO*Z1*W;

Calculate X2=R^2-T*W^2;

Calculate V=T*W^2-2*X2; and

Calculate 2*Y2=V*R-M*W^3.

5. (Amended) A countermeasure method according to Claim 1, wherein the elliptical curve is defined on the finite field GF(2^n), where n is a prime number, and the step of drawing a random integer comprises

Drawing at random a non-zero element l of GF(2^n).

6. (Amended) A countermeasure method according to Claim 1, 5, further including the following steps:

Calculate Z2=X'1*Z'1^2;

Calculate X2=(X'1+c*Z'1^2)^4;

Calculate U=Z2+X'1^2+Y'1*Z'1; and

Calculate Y2=X'1^4*Z2+U*X2.

7. (Amended) A countermeasure method according to Claim 5, further including the following steps, with the calculation in each of the steps being carried out modulo p:

For an input point P=(X0, Y0, Z0), replacing X0 with l^2*X0, Y0 with l^3*Y0 and Z0 with l*Z0;

3) Drawing at random a non-zero element m of GF(2^n);

4) For an input point Q = (X1, Y1, Z1), replacing X1 with m^2*X1, Y1 with m^3*Y1 and Z1 with m*Z1; and

5) Calculating $R=P+Q$.

8. (Amended) A countermeasure method according to Claim 5, further including the following steps:

For an input point $P=(X0, Y0, Z0)$, replacing $X0$ with $l^2*X0$, $Y0$ with $l^3*Y0$ and $Z0$ with $l*Z0$;

Drawing at random a non-zero element m of $GF(2^n)$;

For an input point $Q = (X1, Y1, Z1)$ replacing $X1$ with $m^2*X1$, $Y1$ with $m^3*Y1$ and $Z1$ with $m*Z1$;

Calculate $U0=X0*Z1^2$;

Calculate $S0=Y0*Z1^3$;

Calculate $U1=X1*Z0^2$;

Calculate $S1=Y1*Z0^3$;

Calculate $W=U0+U1$;

Calculate $R=S0+S1$;

Calculate $L=Z0*W$;

Calculate $V=R*X1+L*Y1$;

Calculate $Z2=L*Z1$;

Calculate $T=R+Z2$;

Calculate $X2=a*Z2^2+T*R+W^3$; and

Calculate $Y2=T*X2+V*L^2$.

9. (Amended) A countermeasure method according to Claim 1, further including the process of randomizing the representation of a point at the start of the calculation by the use of a "double and add" algorithm, taking as an input a point P and an integer d, the integer d being denoted $d=(d(t),d(t-1),\ldots,d(0))$, where $(d(t),d(t-1),\ldots,d(0))$ is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit, the algorithm returning as an output the point $Q=d.P$, according to the following steps:

1) Initialising the point Q with the value P;
2) Replacing Q with 2.Q;
3) If $d(t-1)=1$ replacing Q with Q+P;
4) For i ranging from t-2 to 0 executing the steps of:
   - 4a) Replacing Q with 2Q;
   - 4b) If $d(i)=1$, replacing Q with Q+P; and
5) Returning Q.



10. (Amended) A countermeasure method according to Claim 1, further including the process of randomizing the representation of a point at the start of the calculation method and at the end of the calculation method, using a "double and add" algorithm, taking as an input a point P and an integer d, the integer d being denoted $d=(d(t),d(t-1),\ldots,d(0))$, where $(d(t),d(t-1),\ldots,d(0))$ is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit, the algorithm returning as an output the point $Q=d.P$, according to the following steps:

1) Initialising the point Q with the value P;

2) Replacing Q with 2.Q;

3) If d(t-1)=1, replacing Q with Q+P;

4) For i ranging from t-2 to 1, executing the steps of:

4a) Replacing Q with 2Q;

4b) If d(i)=1, replacing Q with Q+P;

5) Replacing Q with 2.Q;

6) If d(0)=1, replacing Q with Q+P and;

7) Returning Q.

11. (Amended) A countermeasure method according to Claim 1, further including the following steps:

1) Initialising the point Q with the point P;

2) For i ranging from t-2 to 0, executing the steps of:

2a) Replacing Q with 2Q;

2b) If d(i)=1, replacing Q with Q+P; and

3) Returning Q.

12. (Amended) A countermeasure method according to Claim 1, further including the following steps:

1) Initialising the point Q with the point P.

2) Initialising a counter co to the value T.

3) For i ranging from t-1 to 0, executing the steps of:

3a) Replacing Q with 2Q using a first method if co is different from 0, otherwise using method;

3b) If d(i)=1, replacing Q with Q+P;

3c) If co=0 then reinitialising the counter co to the value T;

3d) Decrementing the counter co; and

4 Returning Q.

13. (Amended) The method of claim 1, wherein said electronic component is a smart card.

## REMARKS

Entry of the foregoing amendment is respectfully requested. This amendment is intended to place the claims in a more conventional format and eliminate the multiple dependency of the claims.

Respectfully submitted,

BURNS, DOANE, SWECKER & MATHIS, L.L.P.

By: ______________________
James A. LaBarre
Registration No. 28,632

P.O. Box 1404
Alexandria, Virginia 22313-1404
(703) 836-6620

Date: September 26, 2001

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

1. (Amended) A countermeasure method in an electronic component implementing an elliptical curve type public key encryption algorithm [using the representation of the points of the said elliptical curve in projective coordinates, consisting of representing], wherein a point P on the elliptical curve is represented by the projective coordinates (X, Y, Z) such that x=X/Z and y=Y/Z^3, x and y being the coordinates of the point on the elliptical curve in terms of affine coordinates, [the] said curve comprising n elements and being defined on a finite field GF(p), where p [being] is a prime number[, the said curve having] and the curve has the equation y^2=x^3+a*x+b, or defined on a finite field GF(2^n), with the [said] curve having the equation y^2+x*y=x^3+a*x^2+b, where a and b are integer parameters [fixed at the start], the [said] method comprising the steps of:

[choosing a random integer representative from amongst n possible elements in terms of projective coordinates of the elliptical curve and consisting of a modification of the operations of addition of points, doubling of the said points and/or a modification of the scalar multiplication operation, characterised in that the countermeasure applies whatever the method or algorithm, hereinafter denoted A, used for performing the point doubling operation, the method A being replaced by the method A' in three steps, using an input defined by a point P=(X1,Y1,Z1) represented in terms of projective coordinates and an output defined by point Q=(X2,Y2,Z2) represented in terms of projective coordinates such that Q=2.P, of the elliptical curve, the said steps being:]

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

1) Drawing at random an integer l such that 0<l<p;

2) For a point P represented by projective coordinates (X1, Y1, Z1), calculating X'1=l^2*X1, Y'1=l^3*Y1 and Z'1=l*Z1, [X'1, Y'1 and Z'1 defining] to define the coordinates of the point P'=(X'1,Y'1,Z'1); and

3) Calculating an output point Q=2*P' [by means of the algorithm A] that is represented by projective coordinates (X2, Y2, Z2).

2. (Amended) A countermeasure method according to Claim 1, [characterised in that the point doubling algorithm, or operations of doubling points on an] wherein the elliptical curve is defined on the [said] finite field GF(p), [is effected in eight] and the step of calculating Q includes the following steps:

[1) Drawing at random an integer l such that 0<l<p;

2) Calculate X'1=l^2*X1, Y'1=l^3*Y1 and Z'1=l*Z1;

3)] Calculate M=3*X'1^2+a*Z'1^4;

[4)] Calculate Z2=2*Y'1*Z'1;

[5)] Calculate S=4*X'1*Y'1^2;

[6)] Calculate X2=M^2-2*S;

[7)] Calculate T=8*Y'1^4; and

[8)] Calculate Y2=M*(S-X2)-T.

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

3. (Amended) A countermeasure method according to Claim 1, [characterised in that more generally the countermeasure method applies whatever the method denoted hereinafter A used for performing the points addition operation on an] wherein the elliptical curve is defined on the [said] finite field GF(p), [is effected in five] and further including the following steps:

[1)] Drawing at random a non-zero integer l of GF(2^n);

[2)] Replacing X0 with l^2*X0, Y0 with l^3*Y0 and Z0 with l*Z0;

[3)] Drawing at random a non-zero integer m of GF(2^n);

[4)] Replacing X1 with m^2*X1, Y1 with m^3*Y1 and Z1 with m*Z1; and

[5)] Calculating R=P+Q [by means of algorithm A].

4. (Amended) A countermeasure method according to Claim 1, [characterised in that the modification of the point addition algorithm for an elliptical curve defined on the finite field GF(p), where p is a prime number, is as follows:] further including the calculation of the projective coordinates of the point R=(X2,Y2,Z2) such that R=P+Q with P=(X0,Y0,Z0) and Q=(X1,Y1,Z1) [are calculated by] according to the following [method in 16] steps, with the calculations in each of the steps [the calculations] being effected modulo p:

[1) Drawing at random an integer l belonging to the finite field GF(p) such that 0<l<p;

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

2)] Replacing X0 with l^2*X0, Y0 with l^3*Y0 and Z0 with l*Z0;

[3)] Drawing at random an integer m [belonging] such that 0<m<p;

[4)] Replacing X1 with m^2*X1, Y1 with m^3*Y1 and Z1 with m*Z1;

[5)] Calculate U0=X0*Z1^2;

[6)] Calculate S0=Y0*Z1^3;

[7)] Calculate U1=X1*Z0^2;

[8)] Calculate S1=Y1*Z0^3;

[9)] Calculate W=U0-U1;

[10)] Calculate R=S0-S1;

[11)] Calculate T=U0+U1;

[12)] Calculate M=S0+S1;

[13)] Calculate Z2=ZO*Z1*W;

[14)] Calculate X2=R^2-T*W^2;

[15)] Calculate V=T*W^2-2*X2; and

[16)] Calculate 2*Y2=V*R-M*W^3.

5. (Amended) A countermeasure method according to Claim 1, [characterised in that, more generally, the modification of the point addition algorithm for an] wherein the elliptical curve is defined on the finite field GF(2^n), where n is a prime number, and the step of drawing a random integer comprises [is as follows: the projective coordinates of the

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

point P=(X1,Y1,Z1) such that R=P+Q and Q=(X2,Y2,Z2) are calculated by the following method in 3 steps, in each of the steps the calculations being carried out modulo p:

1)] Drawing at random a non-zero element l of GF(2^n)[;

2) Calculating X'1=l^2*X1, Y'1=l^3*Y1 and Z'1=l*Z1, X'1, Y'1 and Z'1 defining the coordinates of the point P'=(X'1,Y'1,Z'1);

3) Calculating Q=2.P' by means of the algorithm A].

6. (Amended) A countermeasure method according to Claim 1, [characterised in that the countermeasure method consists of a modification of the previous method, the new point doubling method for an elliptical curve being defined on the finite field GF(2^n), and consists of the following 6] <u>5, further including the following</u> steps:

[1) Drawing at random a non-zero element l of GF(2^n);

2) Calculate X'1=l^2*X1, Y'1=l^3*Y1, Z'1=l*Z1;

3)] Calculate Z2=X'1*Z'1^2;

[4)] Calculate X2=(X'1+c*Z'1^2)^4;

[5)] Calculate U=Z2+X'1^2+Y'1*Z'1; <u>and</u>

[6)] Calculate Y2=X'1^4*Z2+U*X2.

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

7. (Amended) A countermeasure method according to Claim [1, characterised in that, more generally, the modification of the point addition algorithm for an elliptical curve defined on the finite field GF(2^n), where n is a prime number, is as follows: the projective coordinates of the point P=(X0,Y0,Z0) and Q=(X1,Y1,Z2) at the input and R=(X2,Y2,Z2) are calculated by the following method in 5 steps,] 5, further including the following steps, with the calculation in each of the steps [the calculations] being carried out modulo p:

[1) Drawing at random a non-zero element l of GF(2^n);

2)] For an input point P=(X0, Y0, Z0), replacing X0 with l^2*X0, Y0 with l^3*Y0 and Z0 with l*Z0;

3) Drawing at random a non-zero element m of GF(2^n);

4) For an input point Q = (X1, Y1, Z1), replacing X1 with m^2*X1, Y1 with m^3*Y1 and Z1 with m*Z1; and

5) Calculating R=P+Q [using the algorithm A].

8. (Amended) A countermeasure method according to Claim [1, characterised in that the countermeasure method consists of a modification of the point addition method for an elliptical curve defined on the finite field GF(2^n) and consists of] 5, further including the following [16] steps:

[1) Drawing at random a non-zero element l of GF(2^n);

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

2)] For an input point P=(X0, Y0, Z0), replacing X0 with l^2*X0, Y0 with l^3*Y0 and Z0 with l*Z0;

[3)] Drawing at random a non-zero element m of GF(2^n);

[4)] For an input point Q = (X1, Y1, Z1) replacing X1 with m^2*X1, Y1 with m^3*Y1 and Z1 with m*Z1;

[5)] Calculate U0=X0*Z1^2;

[6)] Calculate S0=Y0*Z1^3;

[7)] Calculate U1=X1*Z0^2;

[8)] Calculate S1=Y1*Z0^3;

[9)] Calculate W=U0+U1;

[10)] Calculate R=S0+S1;

[11)] Calculate L=Z0*W;

[12)] Calculate V=R*X1+L*Y1;

[13)] Calculate Z2=L*Z1;

[14)] Calculate T=R+Z2;

[15)] Calculate X2=a*Z2^2+T*R+W^3; and

[16)] Calculate Y2=T*X2+V*L^2.

9. (Amended) A countermeasure method according to Claim 1, [characterised in that the first variant of a modification of the scalar multiplication operation consists of

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

making random] further including the process of randomizing the representation of a point at the start of the calculation [method] by the use of [the] a “double and add” algorithm, [the modified method of scalar multiplication is as follows in 5 steps,] taking as an input a point P and an integer d, the integer d being denoted d=(d(t),d(t-1),...,d(0)), where (d(t),d(t-1),...,d(0)) is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit, the algorithm returning as an output the point Q=d.P, [the method Do being the points doubling method, the method Do’ being the modified points doubling method according to any one of the preceding claims, this first variant being executed in five] according to the following steps:

1) Initialising the point Q with the value P;

2) Replacing Q with 2.Q [using the method Do’];

3) If d(t-1)=1 replacing Q with Q+P [using the method Ad];

4) For i ranging from t-2 to 0 executing the steps of:

    4a) Replacing Q with 2Q;

    4b) If d(i)=1, replacing Q with Q+P; and

5) Returning Q.

10. (Amended) A countermeasure method according to Claim 1, [characterised in that the second variant of the scalar multiplication operation consists in making random] further including the process of randomizing the representation of a point at the start of the

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

calculation method and at the end of the calculation method, [this in the case of the use of the] using a "double and add" algorithm, [the modified scalar multiplication method being the following one in 7 steps,] taking as an input a point P and an integer d, the integer d being denoted d=(d(t),d(t-1),...,d(0)), where (d(t),d(t-1),...,d(0)) is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit, the algorithm returning as an output the point Q=d.P, [the said second variant being executed in seven] according to the following steps:

1) Initialising the point Q with the value P;

2) Replacing Q with 2.Q [using the method Do'];

3) If d(t-1)=1, replacing Q with Q+P [using the method Ad];

4) For i ranging from t-2 to 1, executing the steps of:

   4a) Replacing Q with 2Q;

   4b) If d(i)=1, replacing Q with Q+P;

5) Replacing Q with 2.Q [using the method Do'];

6) If d(0)=1, replacing Q with Q+P [using the method Ad] and;

7) Returning Q.

11. (Amended) A countermeasure method according to Claim 1, [characterised in that the third variant of the scalar multiplication operation is executed in three] further including the following steps:

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

1) Initialising the point Q with the point P;

2) For i ranging from t-2 to 0, executing the steps of:

2a) Replacing Q with 2Q [using the method Do'];

2b) If d(i)=1, replacing Q with Q+P [using the method Ad', Ad' being the method of addition of the modified points according to the preceding claims]; and

3) Returning Q.

12. (Amended) A countermeasure method according to Claim 1, [characterised in that the fourth variant of the scalar multiplication operation is executed in three] further including the following steps:

1) Initialising the point Q with the point P.

2) Initialising [the] a counter co to the value T.

3) For i ranging from t-1 to 0, executing the steps of:

3a) Replacing Q with 2Q using [the] a first method [Do] if co is different from 0, otherwise using [the] method [Do'.];

3b) If d(i)=1, replacing Q with Q+P [using the method Ad.];

3c) If co=0 then reinitialising the counter co to the value T[.];

3d) Decrementing the counter co[.]; and

[3)] 4 Returning Q.

**Attachment to Preliminary Amendment dated September 26, 2001**

**Marked-up Claims 1-13**

13. (Amended) [An] The method of claim 1, wherein said electronic component [using the method according to any one of the preceding claims, characterised in that it can be] is a smart card.

09/937396
JC16 Rec'd PCT/PTO SEP 26 2001



# COUNTERMEASURE METHODS IN AN ELECTRONIC COMPONENT IMPLEMENTING AN ELLIPTICAL CURVE TYPE PUBLIC KEY ENCRYPTION ALGORITHM

The present invention relates to a countermeasure method in an electronic component using an elliptical curve type public key enciphering algorithm.

In the conventional model of secret key encryption, two persons wishing to communicate by means of a non-secure channel must first agree on a secret enciphering key K. The enciphering function and the deciphering function use the same key K. The drawback of the secret key enciphering system is that the said system requires the prior communication of the key K between the two persons by means of a secure channel, before any enciphered message is sent over the non-secure channel. In practice, it is generally difficult to find a perfectly secure communication channel, particularly if the distance separating the two persons is great. Secure channel means a channel for which it

is impossible to know or modify the information passing over the said channel. Such a secure channel can be implemented by means of a cable connecting two terminals, possessed by the said two persons.

The concept of public key encryption was invented by Whitfield Diffie and Martin Hellman in 1976. Public key encryption makes it possible to resolve the problem of the distribution of the keys over a non-secure channel. The principle of public key encryption consists in using a pair of keys, a public enciphering key and a private deciphering key. It must be unfeasible from the calculation point of view to find the private deciphering key from the public enciphering key. A person A wishing to communicate information to a person B uses the public enciphering key of the person B. Only the person B possesses the private key associated with his public key. Only the person B is therefore capable of deciphering the message sent to him.

Another advantage of public key encryption over secret key encryption is that public key encryption allows authentication by the use of an electronic signature.

The first implementation of the public key enciphering scheme was developed in 1977 by Rivest, Shamir and Adleman, who invented the RSA enciphering system. RSA security is based on the difficulty of factorising a large number which is the product of two prime numbers.

Since then, many public key enciphering systems have been proposed, the security of which is based on different calculatory problems (this list is not exhaustive):

- Merckle-Hellman backpack:

This enciphering system is based on the difficulty of the problem of the sum of subsets.

- McEliece:

This enciphering system is based on the theory of algebraic codes. It is based on the problem of the decoding of linear codes.

- El Gamal:

This enciphering system is based on the difficulty of the discrete logarithm in a finite field.

- Elliptical curves:

The elliptical curve enciphering system constitutes a modification to existing cryptographic systems in order to apply them to the field of elliptical curves.

The use of elliptical curves in cryptographic systems was proposed independently by Victor Miller and Neal Koblitz in 1985. Actual applications of elliptical curves were envisaged early in the 1990s.

The advantage of cryptosystems based on elliptical curves is that they provide security equivalent to other cryptosystems but with smaller key sizes. This saving in key size entails a decrease in memory requirements and a reduction in calculation times, which makes the use of elliptical curves

particularly suitable for applications of the smart card type.

An elliptical curve on a finite field $GF(q^n)$ ($q$ being a prime number and $n$ an integer) is the set of points $(x,y)$ with $x$ the X-axis and $y$ the Y-axis belonging to $GF(q^n)$ the solution to the equation:

$$y^2=x^3+a*x+b$$

if $q$ is greater than or equal to 3 and

$$y^2+x*y=x^3+a*x^2+b$$

if $q=2$.

There are 2 methods for representing a point on an elliptical curve:

Firstly, affine coordinates representation; in this method, a point P on the elliptical curve is represented by its coordinates $(x,y)$.

Secondly, projective coordinates representation.

The advantage of projective coordinates representation is that it makes it possible to avoid divisions in the finite field, the said divisions being the most expensive operations in terms of calculation time.

The most frequently used projective coordinates representation is that consisting of representing a point P on the elliptical curve by the coordinates $(X,Y,Z)$, such that $x=X/Z$ and $y=Y/Z^3$.

The projective coordinates of a point are not unique since the triplet $(X,Y,Z)$ and the triplet $(\lambda^2*X, \lambda^3*Y, \lambda*Z)$ represent the same point whatever the element $\lambda$ belonging to the finite field on which the elliptical curve is defined.

The two classes of curves which are most used in encryption are the following:

1) Curves defined on the finite field GF(p) (the set of integers modulo p, p being a prime number) having the equation:

$$y^2=x^3+a*x+b$$

2) Curves defined on the finite field GF(2^n) having the equation

$$y^2+x*y=x^3+a*x^2+b$$

For each of these two classes of curve, the point addition and point doubling operations are defined.

Point addition is the operation which, given two points P and Q, calculates the sum R=P+Q, R being a point on the curve whose coordinates are expressed by means of the coordinates of the points P and Q in accordance with formulae whose expression is given in the work "Elliptical curve public key cryptosystem" by Alfred J. Menezes.

Point doubling is the operation which, given a point P, calculates the point R=2*P, R being a point on the curve whose coordinates are expressed by means of the coordinates of the point P in accordance with the formulae whose expression is given in the work "Elliptical curve public key cryptosystem" by Alfred J. Menezes.

The point addition and point doubling operations make it possible to define a scalar multiplication operation: given a point P belonging to an elliptical curve and an integer d, the result of the scalar

multiplication of P by d is the point Q such that Q=d*P=P+P+…+P d times.

The security of encryption algorithms on elliptical curves is based on the difficulty of the problem of the discrete logarithm on elliptical curves, the said problem consisting, using two points Q and P belonging to an elliptical curve E, in finding, if such exists, an integer x such that Q=x*P.

There are many cryptographic algorithms based on the problem of the discrete logarithm. These algorithms are easily transposable to elliptical curves.

Thus it is possible to use algorithms providing authentication, confidentiality, integrity check and key exchange.

A point common to the majority of cryptographic algorithms based on elliptical curves is that they comprise as a parameter an elliptical curve defined on a finite field and a point P belonging to this elliptical curve. The private key is an integer d chosen randomly. The public key is a point on the curve Q such that Q=d*P. These cryptographic algorithms generally involve a scalar multiplication in the calculation of a point R=d*T, where d is the secret key.

In the above section, an enciphering algorithm based on an elliptical curve is described. This scheme is similar to the El Gamal enciphering scheme. A message m is enciphered as follows:

The cipher clerk chooses an integer k randomly and calculates the points k*P=(x1,y1) and k*Q=(x2,y2) on the curve, and the integer c= x2 + m. The cipher of m is the triplet (x1,y1,c).

The deciphering clerk, who possesses d, deciphers m by calculating:

(x'2,y'2)=d(x1,y1) and m=c-x'2

In order to effect the scalar multiplications necessary in the calculation methods described previously, several algorithms exist:

"Double and add" algorithm;

"Addition-subtraction" algorithm;

Algorithm with addition chains;

Algorithm with window;

Algorithm with signed representation.

This list is not exhaustive. The simplest algorithm and the one which is most used is the "double and add" algorithm. The "double and add" algorithm takes as its input a point P belonging to a given elliptical curve and an integer d. The integer d is denoted d=(d(t),d(t-1),…,d(0)), where (d(t),d(t-1),…,d(0)) is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit. The algorithm returns as an output the point Q=d.P.

The "double and add" algorithm includes the following three steps:

1) Initialising the point Q with the value P

2) For i ranging from t-1 to 0, executing:

 2a) Replacing Q with 2Q

2b) If d(i)=1 replacing Q with Q+P

3) Returning Q.

It became clear that the implementation of a public key enciphering algorithm of the elliptical curve type on a smart card was vulnerable to attacks consisting of a differential analysis of current consumption making it possible to find the private deciphering key. These attacks are known as DPA attacks, the acronym for Differential Power Analysis. The principle of these DPA attacks is based on the fact that the current consumption of the microprocessor executing the instructions varies according to the data item being manipulated.

In particular, when an instruction is manipulating a data item in which a particular bit is constant, where the value of the other bits may vary, analysis of the current consumption related to the instruction shows that the mean consumption of the instruction is not the same according to whether the particular bit takes the value 0 or 1. The attack of the DPA type therefore makes it possible to obtain additional information on the intermediate data manipulated by the microprocessor of the card when a cryptographic algorithm is being executed. This additional information can in some cases reveal the private parameters of the deciphering algorithm, making the cryptographic system insecure.

In the remainder of this document a description is given of a method of DPA attack on an algorithm of the elliptical curve type performing an operation of

the type consisting of the scalar multiplication of a point P by an integer d, the integer d being the secret key. This attack directly reveals the secret key d. It therefore seriously compromises the security of the implementation of elliptical curves on a smart card.

The first step of the attack is the recording of the current consumption corresponding to the execution of the "double and add" algorithm described previously for N distinct points P(1),…, P(N). In an algorithm based on elliptical curves, the microprocessor of the smart card will perform N scalar multiplications d.P(1),…,d.P(N).

For clarity of the description of the attack, the first step is to describe a method for obtaining the value of the bit d(t-1) of the secret key d, where (d(t),d(t-1),…,d(0)) is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit. Next the description of an algorithm which makes it possible to find the value of d is given.

The points P(1) to P(N) are grouped together according to the value of the last bit of the abscissa of 4.P, where P designates one of the points P(1) to P(N). The first group consists of the points P such that the last bit of the abscissa of 4.P is equal to 1.

The second group consists of the points P such that the last bit of the abscissa of 4.P is equal to 0. The mean of the current consumptions corresponding to each of the two groups is calculated, and the difference curve between these two means is calculated.

If the bit d(t-1) of d is equal to 0, then the scalar multiplication algorithm previously described calculates and stores in memory the value of 4.P. This means that, when the algorithm is executed in a smart card, the microprocessor of the card will actually calculate 4.P. In this case, in the first message group, the last bit of the data item manipulated by the microprocessor is always at 1, and in the second message group the last bit of the data item manipulated is always at 0. The mean of the current consumptions corresponding to each group is therefore different. There therefore appears, in the difference curve between the two means, a differential current consumption peak.

If on the other hand the bit d(t-1) of d is equal to 1, the exponentiation algorithm described previously does not calculate the point 4.P. When the algorithm is executed by the smart card, the microprocessor therefore never manipulates the data item 4.P. Therefore no differential consumption peak appears.

This method therefore makes it possible to determine the value of the bit d(t-1) of d.

The algorithm described in the following section is a generalisation of the previous algorithm. It makes it possible to determine the value of the secret key d.

The input is defined by N points denoted P(1) to P(N) corresponding to N calculations performed by the smart card, and the output by an integer h.

The said algorithm is implemented as follows in three steps.

1) Executing h=1;

2) For i ranging from t-1 to 1, executing:

2)1) Classifying the points P(1) to P(N) according to the value of the last bit of the abscissa of (4*h).P;

2)2) Calculating the current consumption mean for each of the two groups;

2)3) Calculating the difference between the two means;

2)4) If the difference shows a differential consumption peak, doing h=h*2; otherwise doing h=h*2+1;

3) Returning h.

The above algorithm supplies an integer h such that d=2*h or d=2*h+1. In order to obtain the value of d, it then suffices to test the two possible hypotheses.

The attack of the DPA type described therefore makes it possible to find the private key d.

The method of the invention consists in devising of a countermeasure for guarding against the DPA attack described above. This countermeasure uses the representation of the points on the elliptical curve in projective coordinates.

As explained above, the representative of a point in projective coordinates is not unique. If the finite field on which the elliptical curve is defined

comprises n elements, it is possible to choose one representative amongst n-1 possible ones.

By choosing a random representative of a point on which the calculation is carried out, the intermediate values of the calculation themselves become random and therefore unpredictable from outside, which makes the DPA attack described above impossible.

The countermeasure method consists of a modification of the elliptical curve point doubling and point addition operations defined on the finite fields GF(p) for p prime and GF(2^n). The modification of the point addition and point doubling operations on elliptical curves defined on the finite fields GF(p) for p prime and GF(2^n) apply whatever the algorithm used for performing these operations.

The countermeasure method also consists of the definition of four variants in the scalar multiplication operation. These four variants apply whatever the algorithm used for performing the scalar multiplication operation.

In this section, a description is given of the modification of the point doubling algorithm for an elliptical curve defined on the finite field GF(p), where p is a prime number. The elliptical curve is therefore defined by the following equation:

y^2=x^3+a*x+b

where a and b are integer parameters fixed at the start.

The projective coordinates of the point Q=(X2,Y2,Z2) such that Q=2.P with P=(X1,Y1,Z1) are

calculated by the following method in 6 steps. In each of the steps, the calculations are effected modulo p.

1) Calculate M=3*X1^2+a*Z1^4;

2) Calculate Z2=2*Y1*Z1;

3) Calculate S=4*X1*Y1^2;

4) Calculate X2=M^2=2*S;

5) Calculate T=8*Y1^4;

6) Calculate Y2=M*(S-X2)-T.

The countermeasure method consists of a modification of the above method.

The new method of point doubling for an elliptical curve defined on the finite field GF(p) consists of the following 8 steps:

1) Drawing at random an integer λ such that 0<λ<p;

2) Calculate X'1=λ^2*X1, Y'1=λ^3*Y1 and Z'1=λ*Z1;

3) Calculate M=3*X'1^2+a*Z'1^4;

4) Calculate Z2=2*Y'1*Z'1;

5) Calculate S=4*X'1*Y'1^2;

6) Calculate X2=M^2-2*S;

7) Calculate T=8*Y'1^4;

8) Calculate Y2=M*(S-X2)-T.

More generally, the countermeasure method applies whatever the method (hereinafter denoted A) used for performing the point doubling operation. The method A is replaced by the method A' in 3 steps:

Input: a point P=(X1,Y1,Z1) represented in projective coordinates.

Output: a point Q=(X2,Y2,Z2) represented in projective coordinates such that Q=2.P.

1) Drawing at random an integer $\lambda$ such that $0<\lambda<p$;

2) Calculating $X'1=\lambda^2*X1$, $Y'1=\lambda^3*Y1$ and $Z'1=\lambda*Z1$, $X'1$, $Y'1$ and $Z'1$ defining the coordinates of the point $P'=(X'1,Y'1,Z'1)$;

3) Calculating $Q=2*P'$ by means of the algorithm A.

The variables manipulated during the execution of the method A' being random, the previously described DPA attack no longer applies.

In this paragraph, a description is given of the modification to the point addition algorithm for an elliptical curve defined on the finite field GF(p), where p is a prime number.

The projective coordinates of the point $R=(X2,Y2,Z2)$ such that $R=P+Q$ with $P=(X0,Y0,Z0)$ and $Q=(X1,Y1,Z1)$ are calculated by the following method in 12 steps. In each of the steps, the calculations are carried out modulo p.

1) Calculate $U0=X0*Z1^2$;

2) Calculate $S0=Y0*Z1^3$;

3) Calculate $U1=X1*Z0^2$;

4) Calculate $S1=Y1*Z0^3$;

5) Calculate $W=U0-U1$;

6) Calculate $R=S0-S1$;

7) Calculate $T=U0+U1$;

8) Calculate $M=S0+S1$;

9) Calculate $Z2=ZO*Z1*W$;

10) Calculate $X2=R^2-T*W^2$;

11) Calculate V=T*W^2-2*X2;

12) Calculate 2*Y2=V*R-M*W^3.

The countermeasure method consists of a modification of the previous method. The new method of point addition for an elliptical curve defined on the finite field GF(p) consists of the following 16 steps:

1) Drawing at random an integer $\lambda$ such that $0<\lambda<p$;

2) Replacing X0 with $\lambda$^2*X0, Y0 with $\lambda$^3*Y0 and Z0 with $\lambda$*Z0;

3) Drawing at random an integer $\mu$ such that $0<\mu<p$;

4) Replacing X1 with $\mu$^2*X1, Y1 with $\mu$^3*Y1 and Z1 with $\mu$*Z1;

5) Calculate U0=X0*Z1^2;

6) Calculate S0=Y0*Z1^3;

7) Calculate U1=X1*Z0^2;

8) Calculate S1=Y1*Z0^3;

9) Calculate W=U0-U1;

10) Calculate R=S0-S1;

11) Calculate T=U0+U1;

12) Calculate M=S0+S1;

13) Calculate Z2=ZO*Z1*W;

14) Calculate X2=R^2-T*W^2;

15) Calculate V=T*W^2-2*X2;

16) Calculate 2*Y2=V*R-M*W^3.

More generally, the countermeasure method applies whatever the method (hereinafter denoted A) used for

performing the point addition operation. The method A is replaced by the method A′ in 5 steps:

Input: two points P=(X0,Y0,Z0) and Q=(X1,Y1,Z1) represented in projective coordinates.

Output: the point R=(X2,Y2,Z2) represented in projective coordinates such that R=P+Q.

1) Drawing at random an integer $\lambda$ such that $0<\lambda<p$;

2) Replacing X0 with $\lambda$^2*X0, Y0 with $\lambda$^3*Y0 and Z0 with $\lambda$*Z0;

3) Drawing at random an integer $\mu$ such that $0<\mu<p$;

4) Replacing X1 with $\mu$^2*X1, Y1 with $\mu$^3*Y1 and Z1 with $\mu$*Z1;

5) Calculating R=P+Q by means of algorithm A.

The variables manipulated during the execution of the method A′ being random, the previously described DPA attack no longer applies.

In this section, a description is given of the modification of the point doubling algorithm for an elliptical curve defined on the finite field GF(2^n). The elliptical curve is therefore defined by the following equation:

y^2+x*y=x^3+a*x^2+b

where a and b are parameters belonging to the finite field GF(2^n) fixed at the start. c is defined by the equation:

c=b^(2^(n-2)).

The projective coordinates of the point Q=(X2,Y2,Z2) such that Q=2.P with P=(X1,Y1,Z1) are calculated by the following method in 4 steps. In each of the steps, the calculations are carried out in the finite field GF(2^n).

1) Calculate Z2=X1*Z1^2;

2) Calculate X2=(X1+c*Z1^2)^4;

3) Calculate U=Z2+X1^2+Y1*Z1;

4) Calculate Y2=X1^4*Z2+U*X2.

The countermeasure method consists of a modification of the previous method. The new point doubling method for an elliptical curve defined on the finite field GF(2^n) consists of the following 6 steps:

1) Drawing at random a non-zero element $\lambda$ of GF(2^n);

2) Calculate X'1=$\lambda$^2*X1, Y'1=$\lambda$^3*Y1, Z'1=$\lambda$*Z1;

3) Calculate Z2=X'1*Z'1^2;

4) Calculate X2=(X'1+c*Z'1^2)^4;

5) Calculate U=Z2+X'1^2+Y'1*Z'1;

6) Calculate Y2=X'1^4*Z2+U*X2.

More generally, the countermeasure method applies whatever the method (hereinafter denoted A) used for performing the point doubling operation. The method A is replaced by the method A' in 3 steps:

Input: a point P=(X1,Y1,Z1) represented in projective coordinates.

Output: a point Q=(X2,Y2,Z2) represented in projective coordinates such that Q=2.P.

1) Drawing at random a non-zero element $\lambda$ of GF(2^n);

2) Calculating X'1=λ^2*X1, Y'1=λ^3*Y1, Z'1=λ*Z1, X'1, Y'1 and Z'1 defining the coordinates of the point P'=(X'1,Y'1,Z'1);

3) Calculation of Q=2.P' using the algorithm A. The variables manipulated during the execution of the method A' being random, the previously described DPA attack no longer applies.

In this section, a description is given of the modification of the point addition algorithm for an elliptical curve defined on the finite field GF(2^n).

The projective coordinates of the point R=(X2,Y2,Z2) such that R=P+Q with P=(X0,Y0,Z0) and Q=(X1,Y1,Z1) are calculated by the following method in 12 steps. In each of the steps, the calculations are carried out in the finite field GF(2^n).

1) Calculate U0=X0*Z1^2;
2) Calculate S0=Y0*Z1^3;
3) Calculate U1=X1*Z0^2;
4) Calculate S1=Y1*Z0^3;
5) Calculate W=U0+U1;
6) Calculate R=S0+S1;
7) Calculate L=Z0*W;
8) Calculate V=R*X1+L*Y1;
9) Calculate Z2=L*Z1;
10) Calculate T=R+Z2;
11) Calculate X2=a*Z2^2+T*R+W^3;
12) Calculate Y2=T*X2+V*L^2.

The countermeasure method consists of a modification to the previous method. The new point addition method for an elliptical curve defined on the

finite field GF(2^n) consists of the following 14 steps:

1) Drawing at random a non-zero element $\lambda$ of GF(2^n);

2) Replacing X0 with $\lambda$^2*X0, Y0 with $\lambda$^3*Y0 and Z0 with $\lambda$*Z0;

3) Drawing at random a non-zero element $\mu$ of GF(2^n);

4) Replacing X1 with $\mu$^2*X1, Y1 with $\mu$^3*Y1 and Z1 with $\mu$*Z1;

5) Calculate U0=X0*Z1^2;

6) Calculate S0=Y0*Z1^3;

7) Calculate U1=X1*Z0^2;

8) Calculate S1=Y1*Z0^3;

9) Calculate W=U0+U1;

10) Calculate R=S0+S1;

11) Calculate L=Z0*W;

12) Calculate V=R*X1+L*Y1;

13) Calculate Z2=L*Z1;

14) Calculate T=R+Z2;

15) Calculate X2=a*Z2^2+T*R+W^3;

16) Calculate Y2=T*X2+V*L^2.

More generally, the countermeasure method applies whatever the method (hereinafter denoted A) used for performing the point addition operation. The method A is replaced by the method A′ in 5 steps:

Input: two points P=(X0,Y0,Z0) and Q=(X1,Y1,Z1) represented as projective coordinates.

Output: the point R=(X2,Y2,Z2) represented as projective coordinates such that R=P+Q.

1) Drawing at random a non-zero element $\lambda$ of GF(2^n);

2) Replacing X0 with $\lambda$^2*X0, Y0 with $\lambda$^3*Y0 and Z0 with $\lambda$*Z0;

3) Drawing at random a non-zero element $\mu$ of GF(2^n);

4) Replacing X1 with $\mu$^2*X1, Y1 with $\mu$^3*Y1 and Z1 with $\mu$*Z1;

5) Calculating R=P+Q by means of the algorithm A.

The variables manipulated during the execution of the method A' being random, the previously described DPA attack no longer applies.

The countermeasure method also consists in defining four variants in the scalar multiplication operation. The scalar multiplication operation uses the point doubling operation denoted Do and the point addition operation denoted Ad. The modified point doubling operation described above is denoted Do' and the modified point addition operation described above is denoted Ad'.

In this section a description is given of the first variation of the modification to the scalar multiplication operation. The first variant consists of making random the representation of a point at the start of the calculation method. In the case of the use of the "double and add" algorithm, the modified

scalar multiplication method is the following one in 5 steps. The method takes as an input a point P and an integer d. The integer d is denoted d=(d(t),d(t-1),…,d(0)), where (d(t),d(t-1),…,d(0)) is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit. The algorithm returns the point Q=d.P as an output.

This first variant is executed in five steps.

1) Initialising the point Q with the value P;

2) Replacing Q with 2.Q using the method Do';

3) If d(t-1)=1 replacing Q with Q+P using the method Ad;

4) For i ranging from t-2 to 0 executing:

4a) Replacing Q with 2Q;

4b) If d(i)=1, replacing Q with Q+P;

5) Returning Q.

More generally, the method of the first variant described previously applies to the scalar multiplication operation whatever the method (hereinafter denoted A) used for effecting the calculation of the scalar multiplication. The method A uses the previously defined operations Do and Ad.

The first variant of the countermeasure consists in replacing the first operation Do with Do' defined previously.

The first variant therefore ensures that the intermediate variables manipulated during the scalar multiplication operation are random. This makes the previously described DPA attack inapplicable.

In this paragraph the second variant of modification of the scalar multiplication operation is described.

The second variant consists in making random the representation of a point at the start of the calculation method and at the end of the calculation method. In the case of the use of the "double and add" algorithm, the modified scalar multiplication method is the following one in 7 steps. The method takes as an input a point P and an integer d. The integer d is denoted d=(d(t),d(t-1),…,d(0)), where (d(t),d(t-1),…,d(0)) is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit. The algorithm returns the point Q=d.P as an output.

This second variant is executed in seven steps:

1) Initialising the point Q with the value P;

2) Replacing Q with 2.Q using the method Do';

3) If d(t-1)=1, replacing Q with Q+P using the method Ad;

4) For i ranging from t-2 to 1, executing:

4a) Replacing Q with 2Q;

4b) If d(i)=1, replacing Q with Q+P;

5) Replacing Q with 2.Q using the method Do';

6) If d(0)=1, replacing Q with Q+P using the method Ad;

7) Returning Q.

More generally, the method of the second variant described previously applies to the scalar multiplication operation whatever the method

(hereinafter denoted A) used for effecting the calculation of this scalar multiplication. The method A uses the operations Do and Ad defined previously. The second variant of the countermeasure consists of replacing the first operation Do with Do′ defined previously and the last operation Do with Do′.

The second variant therefore ensures that the intermediate variables manipulated during the scalar multiplication operation are random. The advantage of the second variant is increased security against DPA attacks at the end of the scalar multiplication algorithm. In particular, the second variant makes the previously described DPA attack inapplicable.

In this section, the third variant of the modification of the scalar multiplication operation is described.

The third variant consists in making random the representation of each of the points manipulated during the scalar multiplication method. In the case of the use of the "double and add" algorithm, the modified scalar multiplication method is the following one in 4 steps. The method takes as an input a point P and an integer d. The integer d is denoted d=(d(t),d(t-1),…,d(0)), where (d(t),d(t-1),…,d(0)) is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit. The algorithm returns the point Q=d.P as an output.

This third variant is executed in three steps:

1) Initialising the point Q with the point P;

2) For i ranging from t-2 to 0, executing:

2a) Replacing Q with 2Q using the method Do';

2b) If d(i)=1, replacing Q with Q+P using the method Ad';

3) Returning Q.

More generally, the method of the third variant described above applies to the scalar multiplication operation whatever the method (hereinafter denoted A) used for performing the calculation of the scalar multiplication. The method A uses the previously defined operations Do and Ad.

The third variant of the countermeasure consists of replacing all the operations Do with Do' and Ad with Ad'.

The third variant therefore ensures that the intermediate variables manipulated during the scalar multiplication operation are random. The advantage of the third variant compared with the second variant is increased security against DPA attacks on the intermediate operations of the scalar multiplication method. In particular, the third variant makes the previously described DPA attack inapplicable.

In this section the fourth variant of modification of the scalar multiplication operation is described. The fourth variant consists in making random the representation of each of the points manipulated during the scalar multiplication method. The fourth variant is a modification of the third variant through the use of a counter, the said counter making it possible to determine the steps of the scalar multiplication algorithm for which the representation

of a point is made random. For this purpose a security parameter T is defined. In practice T=5 can be taken. In the case of the use of the "double and add" algorithm, the modified scalar multiplication method is the following one in 4 steps. The method takes as an input a point P and an integer d.

The integer d is denoted d=(d(t),d(t-1),…,d(0)), where (d(t),d(t-1),…,d(0)) is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit. The algorithm returns as an output the point Q=d.P.

The fourth variant is executed in three steps:

1) Initialising the point Q with the point P.

2) Initialising the counter co to the value T.

3) For i ranging from t-1 to 0, executing:

3a) Replacing Q with 2Q using the method Do if co is different from 0, otherwise using the method Do'.

3b) If d(i)=1, replacing Q with Q+P using the method Ad.

3c) If co=0 then reinitialising the counter co to the value T.

3d) Decrementing the counter co.

3) Returning Q.

More generally, the method of the third variant described above applies to the scalar multiplication operation whatever the method (hereinafter denoted A) used for effecting the calculation of the scalar multiplication. The method A uses the previously defined operations Do and Ad.

The variant of the third countermeasure consists in initialising a counter co to the value T. The operation Do is replaced by the operation Do' if the value of the counter is 0.

After each execution of the operations Do or Do', the counter is reinitialised to the value T if it has reached the value 0; it is then decremented.

The fourth variant therefore ensures that the intermediate variables manipulated during the scalar multiplication operation are random. The advantage of the fourth variant compared with the third variant is a greater speed of execution. The fourth variant makes the previously described DPA attack inapplicable.

The application of one of the four variants described above therefore makes it possible to protect any cryptographic algorithm based on elliptical curves against the previously described DPA attack.

CLAIMS

1. A countermeasure method in an electronic component implementing an elliptical curve type public key encryption algorithm using the representation of the points of the said elliptical curve in projective coordinates, consisting of representing a point P on the elliptical curve by the coordinates (X, Y, Z) such that x=X/Z and y=Y/Z^3, x and y being the coordinates of the point on the elliptical curve in terms of affine coordinates, the said curve comprising n elements and being defined on a finite field GF(p), p being a prime number, the said curve having the equation y^2=x^3+a*x+b, or defined on a finite field GF(2^n), the said curve having the equation y^2+x*y=x^3+a*x^2+b, where a and b are integer parameters fixed at the start, the said method choosing a random integer representative from amongst n possible elements in terms of projective coordinates of the elliptical curve and consisting of a modification of the operations of addition of points, doubling of the said points and/or a modification of the scalar multiplication operation, characterised in that the countermeasure applies whatever the method or algorithm, hereinafter denoted A, used for performing the point doubling operation, the method A being replaced by the method A' in three steps, using an input defined by a point P=(X1,Y1,Z1) represented in terms of projective coordinates and an output defined by point Q=(X2,Y2,Z2) represented in

terms of projective coordinates such that $Q=2.P$, of the elliptical curve, the said steps being:

1) Drawing at random an integer $\lambda$ such that $0<\lambda<p$;

2) Calculating $X'1=\lambda^2*X1$, $Y'1=\lambda^3*Y1$ and $Z'1=\lambda*Z1$, $X'1$, $Y'1$ and $Z'1$ defining the coordinates of the point $P'=(X'1,Y'1,Z'1)$;

3) Calculating $Q=2*P'$ by means of the algorithm A.

2. A countermeasure method according to Claim 1, characterised in that the point doubling algorithm, or operations of doubling points on an elliptical curve defined on the said finite field GF(p), is effected in eight steps:

1) Drawing at random an integer $\lambda$ such that $0<\lambda<p$;

2) Calculate $X'1=\lambda^2*X1$, $Y'1=\lambda^3*Y1$ and $Z'1=\lambda*Z1$;

3) Calculate $M=3*X'1^2+a*Z'1^4$;

4) Calculate $Z2=2*Y'1*Z'1$;

5) Calculate $S=4*X'1*Y'1^2$;

6) Calculate $X2=M^2-2*S$;

7) Calculate $T=8*Y'1^4$;

8) Calculate $Y2=M*(S-X2)-T$.

3. A countermeasure method according to Claim 1, characterised in that more generally the countermeasure method applies whatever the method denoted hereinafter A used for performing the points addition operation on an elliptical curve defined on the said finite field GF(p) is effected in five steps:

1) Drawing at random a non-zero integer $\lambda$ of GF(2^n);

2) Replacing X0 with $\lambda$^2*X0, Y0 with $\lambda$^3*Y0 and Z0 with $\lambda$*Z0;

3) Drawing at random a non-zero integer $\mu$ of GF(2^n);

4) Replacing X1 with $\mu$^2*X1, Y1 with $\mu$^3*Y1 and Z1 with $\mu$*Z1;

5) Calculating R=P+Q by means of algorithm A.

4. A countermeasure method according to Claim 1, characterised in that the modification of the point addition algorithm for an elliptical curve defined on the finite field GF(p), where p is a prime number, is as follows: the projective coordinates of the point R=(X2,Y2,Z2) such that R=P+Q with P=(X0,Y0,Z0) and Q=(X1,Y1,Z1) are calculated by the following method in 16 steps, in each of the steps the calculations being effected modulo p:

1) Drawing at random an integer $\lambda$ belonging to the finite field GF(p) such that $0<\lambda<p$;

2) Replacing X0 with $\lambda$^2*X0, Y0 with $\lambda$^3*Y0 and Z0 with $\lambda$*Z0;

3) Drawing at random an integer $\mu$ belonging such that $0<\mu<p$;

4) Replacing X1 with $\mu$^2*X1, Y1 with $\mu$^3*Y1 and Z1 with $\mu$*Z1;

5) Calculate U0=X0*Z1^2;

6) Calculate S0=Y0*Z1^3;

7) Calculate $U1=X1*Z0^2$;

8) Calculate $S1=Y1*Z0^3$;

9) Calculate $W=U0-U1$;

10) Calculate $R=S0-S1$;

11) Calculate $T=U0+U1$;

12) Calculate $M=S0+S1$;

13) Calculate $Z2=ZO*Z1*W$;

14) Calculate $X2=R^2-T*W^2$;

15) Calculate $V=T*W^2-2*X2$;

16) Calculate $2*Y2=V*R-M*W^3$.

5. A countermeasure method according to Claim 1, characterised in that, more generally, the modification of the point addition algorithm for an elliptical curve defined on the finite field GF(2^n), where n is a prime number, is as follows: the projective coordinates of the point $P=(X1,Y1,Z1)$ such that $R=P+Q$ and $Q=(X2,Y2,Z2)$ are calculated by the following method in 3 steps, in each of the steps the calculations being carried out modulo p:

1) Drawing at random a non-zero element $\lambda$ of GF(2^n);

2) Calculating $X'1=\lambda^2*X1$, $Y'1=\lambda^3*Y1$ and $Z'1=\lambda*Z1$, $X'1$, $Y'1$ and $Z'1$ defining the coordinates of the point $P'=(X'1,Y'1,Z'1)$;

3) Calculating $Q=2.P'$ by means of the algorithm A.

6. A countermeasure method according to Claim 1, characterised in that the countermeasure method consists of a modification of the previous method, the new point doubling method for an elliptical curve being

defined on the finite field GF(2^n), and consists of the following 6 steps:

1) Drawing at random a non-zero element λ of GF(2^n);

2) Calculate X'1=λ^2*X1, Y'1=λ^3*Y1, Z'1=λ*Z1;

3) Calculate Z2=X'1*Z'1^2;

4) Calculate X2=(X'1+c*Z'1^2)^4;

5) Calculate U=Z2+X'1^2+Y'1*Z'1;

6) Calculate Y2=X'1^4*Z2+U*X2.

7. A countermeasure method according to Claim 1, characterised in that, more generally, the modification of the point addition algorithm for an elliptical curve defined on the finite field GF(2^n), where n is a prime number, is as follows: the projective coordinates of the point P=(X0,Y0,Z0) and Q=(X1,Y1,Z2) at the input and R=(X2,Y2,Z2) are calculated by the following method in 5 steps, in each of the steps the calculations being carried out modulo:

1) Drawing at random a non-zero element λ of GF(2^n);

2) Replacing X0 with λ^2*X0, Y0 with λ^3*Y0 and Z0 with λ*Z0;

3) Drawing at random a non-zero element μ of GF(2^n);

4) Replacing X1 with μ^2*X1, Y1 with μ^3*Y1 and Z1 with μ*Z1;

5) Calculating R=P+Q using the algorithm A.

8. A countermeasure method according to Claim 1, characterised in that the countermeasure method

consists of a modification of the point addition method for an elliptical curve defined on the finite field GF(2^n) and consists of the following 16 steps:

1) Drawing at random a non-zero element $\lambda$ of GF(2^n);

2) Replacing X0 with $\lambda$^2*X0, Y0 with $\lambda$^3*Y0 and Z0 with $\lambda$*Z0;

3) Drawing at random a non-zero element $\mu$ of GF(2^n);

4) Replacing X1 with $\mu$^2*X1, Y1 with $\mu$^3*Y1 and Z1 with $\mu$*Z1;

5) Calculate U0=X0*Z1^2;

6) Calculate S0=Y0*Z1^3;

7) Calculate U1=X1*Z0^2;

8) Calculate S1=Y1*Z0^3;

9) Calculate W=U0+U1;

10) Calculate R=S0+S1;

11) Calculate L=Z0*W;

12) Calculate V=R*X1+L*Y1;

13) Calculate Z2=L*Z1;

14) Calculate T=R+Z2;

15) Calculate X2=a*Z2^2+T*R+W^3;

16) Calculate Y2=T*X2+V*L^2.

9. A countermeasure method according to Claim 1, characterised in that the first variant of a modification of the scalar multiplication operation consists of making random the representation of a point at the start of the calculation method by the use of the "double and add" algorithm, the modified method of

scalar multiplication is as follows in 5 steps, taking as an input a point P and an integer d, the integer d being denoted d=(d(t),d(t-1),…,d(0)), where (d(t),d(t-1),…,d(0)) is the binary representation of d, with d(t) the most significant bit and d(0) the least significant bit, the algorithm returning as an output the point Q=d.P, the method Do being the points doubling method, the method Do' being the modified points doubling method according to any one of the preceding claims, this first variant being executed in five steps:

1) Initialising the point Q with the value P;

2) Replacing Q with 2.Q using the method Do';

3) If d(t-1)=1 replacing Q with Q+P using the method Ad;

4) For i ranging from t-2 to 0 executing:

4a) Replacing Q with 2Q;

4b) If d(i)=1, replacing Q with Q+P;

5) Returning Q.

10. A countermeasure method according to Claim 1, characterised in that the second variant of the scalar multiplication operation consists in making random the representation of a point at the start of the calculation method and at the end of the calculation method, this in the case of the use of the "double and add" algorithm,

the modified scalar multiplication method being the following one in 7 steps, taking as an input a point P and an integer d, the integer d being denoted d=(d(t),d(t-1),…,d(0)), where (d(t),d(t-1),…,d(0)) is the binary representation of d, with d(t) the most

significant bit and d(0) the least significant bit, the algorithm returning as an output the point Q=d.P, the said second variant being executed in seven steps:

1) Initialising the point Q with the value P;

2) Replacing Q with 2.Q using the method Do';

3) If d(t-1)=1, replacing Q with Q+P using the method Ad;

4) For i ranging from t-2 to 1, executing:

4a) Replacing Q with 2Q;

4b) If d(i)=1, replacing Q with Q+P;

5) Replacing Q with 2.Q using the method Do';

6) If d(0)=1, replacing Q with Q+P using the method Ad;

7) Returning Q.

11. A countermeasure method according to Claim 1, characterised in that the third variant of the scalar multiplication operation is executed in three steps:

1) Initialising the point Q with the point P;

2) For i ranging from t-2 to 0, executing:

2a) Replacing Q with 2Q using the method Do';

2b) If d(i)=1, replacing Q with Q+P using the method Ad', Ad' being the method of addition of the modified points according to the preceding claims;

3) Returning Q.

12. A countermeasure method according to Claim 1, characterised in that the fourth variant of the scalar multiplication operation is executed in three steps:

1) Initialising the point Q with the point P.

2) Initialising the counter co to the value T.

3) For i ranging from t-1 to 0, executing:

3a) Replacing Q with 2Q using the method Do if co is different from 0, otherwise using the method Do'.

3b) If d(i)=1, replacing Q with Q+P using the method Ad.

3c) If co=0 then reinitialising the counter co to the value T.

3d) Decrementing the counter co.

3) Returning Q.

13. An electronic component using the method according to any one of the preceding claims, characterised in that it can be a smart card.

OIPE JC67 DEC 0 6 2001 PATENT & TRADEMARK OFFICE

| COMBINED DECLARATION FOR PATENT APPLICATION AND POWER OF ATTORNEY (Includes Reference to Provisional and International (PCT) Applications) | Attorney's Docket No. 032326-169 |
|---|---|

As a below named inventor, I hereby declare that:
My residence, post office address and citizenship are as stated below next to my name;
I BELIEVE I AM THE ORIGINAL, FIRST AND SOLE INVENTOR (IF ONLY ONE NAME IS LISTED BELOW) OR AN ORIGINAL, FIRST AND JOINT INVENTOR (IF PLURAL NAMES ARE LISTED BELOW) OF THE SUBJECT MATTER WHICH IS CLAIMED AND FOR WHICH A PATENT IS SOUGHT ON THE INVENTION ENTITLED:

OIPE DEC 0 6 2001 PATENT & TRADEMARK OFFICE

**COUNTERMEASURE PROCEDURES IN AN ELECTRONIC COMPONENT IMPLEMENTING AN ELLIPTICAL CURVE TYPE PUBLIC KEY ENCRYPTION ALGORITHM**

The specification of which (check only one item below):

☐ is attached hereto.

☐ was filed as United States Patent Application Number ________
on ________
and was amended on ________ (if applicable).

☒ was filed as International (PCT) Application Number **PCT/FR00/00603**
on **March 13, 2000**
and was amended on ________ (if applicable).

I HAVE REVIEWED AND UNDERSTAND THE CONTENTS OF THE ABOVE-IDENTIFIED SPECIFICATION, INCLUDING THE CLAIMS, AS AMENDED BY ANY AMENDMENT REFERRED TO ABOVE.

I ACKNOWLEDGE THE DUTY TO DISCLOSE TO THE U.S. PATENT AND TRADEMARK OFFICE ALL INFORMATION KNOWN TO ME TO BE MATERIAL TO PATENTABILITY AS DEFINED IN TITLE 37, CODE OF FEDERAL REGULATIONS, Sec. 1.56 (as amended effective March 16, 1992);

I do not know and do not believe the said invention was ever known or used in the United States of America before my or our invention thereof, or patented or described in any printed publication in any country before my or our invention thereof or more than one year prior to said application; that said invention was not in public use or on sale in the United States of America more than one year prior to said application; that said invention has not been patented or made the subject of an inventor's certificate issued before the date of said application in any country foreign to the United States of America on any application filed by me or my legal representatives or assigns more than six months prior to said application;

I hereby claim foreign priority benefits under Title 35, United States Code, §§ 119 (a)-(e) of any foreign application(s) for patent or inventor's certificate or of any International (PCT) Application(s) designating at least one country other than the United States of America listed below and have also identified below any foreign application(s) for patent or inventor's certificate or any PCT International (PCT) Application(s) designating at least one country other than the United States of America filed by me on the same subject matter having a filing date before that of the application(s) of which priority is claimed:

PRIOR FOREIGN/PCT APPLICATION(S) AND ANY PRIORITY CLAIMS UNDER 35 U.S.C. §119:

| COUNTRY (if PCT, indicate "PCT") | APPLICATION NUMBER | DATE OF FILING (day, month, year) | PRIORITY CLAIMED UNDER 35 U.S.C. §119 |
|---|---|---|---|
| FRANCE | 99/03921 | March 26, 1999 | ☒Yes ☐No |
| | | | ☐Yes ☐No |
| | | | ☐Yes ☐No |
| | | | ☐Yes ☐No |
| | | | ☐Yes ☐No |

I hereby claim the benefit under Title 35, United States Code § 119(e) of any United States provisional application(s) listed below.

________ (APPLICATION NUMBER) ________ (FILING DATE)

________ (APPLICATION NUMBER) ________ (FILING DATE)

| COMBINED DECLARATION FOR PATENT APPLICATION AND POWER OF ATTORNEY (CONT'D) (Includes Reference to Provisional and International (PCT) Applications) | Attorney's Docket No. 032326-169 |
|---|---|

I hereby claim the benefit under Title 35, United States Code, § 120 of any United States applications(s) or International (PCT) Application(s) designating the United States of America that is/are listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in that/those prior application(s) in the manner provided by the first paragraph of Title 35, United States Code, § 112, I acknowledge the duty to disclose to the U.S. Patent and Trademark Office all information known to me to be material to the patentability as defined in Title 37, Code of Federal Regulations § 1.56, which became available between the filing date of the prior application(s) and the national or international filing date of this application:

PRIOR U.S. APPLICATIONS OR INTERNATIONAL (PCT) APPLICATIONS DESIGNATING THE U.S. FOR BENEFIT UNDER 35 U.S.C. § 120:

| U.S. APPLICATIONS | | | STATUS (check one) | | |
|---|---|---|---|---|---|
| U.S. APPLICATION NUMBER | U.S. FILING DATE | | PATENTED | PENDING | ABANDONED |
| | | | ☐ | ☐ | ☐ |
| | | | | | |
| | | | | | |
| PCT APPLICATIONS DESIGNATING THE U.S. | | | | | |
| PCT APPLICATION NO. | PCT FILING DATE | U.S. APPLICATION NUMBERS ASSIGNED (if any) | | | |
| PCT/FR00/00603 | March 13, 2000 | | | | |
| | | | | | |
| | | | | | |

I hereby appoint the following attorneys and agent(s) to prosecute said application and to transact all business in the U.S. Patent and Trademark Office connected therewith and to file, prosecute and to transact all business in connection with international applications directed to said invention:

39

| | | | | | |
|---|---|---|---|---|---|
| William L. Mathis | 17,337 | R. Danny Huntington | 27,903 | Gerald F. Swiss | 30,113 |
| Robert S. Swecker | 19,885 | Eric H. Weisblatt | 30,505 | Charles F. Wieland III | 33,096 |
| Platon N. Mandros | 22,124 | James W. Peterson | 26,057 | Bruce T. Wieder | 33,815 |
| Benton S. Duffett, Jr. | 22,030 | Teresa Stanek Rea | 30,427 | Todd R. Walters | 34,040 |
| Norman H. Stepno | 22,716 | Robert E. Krebs | 25,885 | Ronni S. Jillions | 31,979 |
| Ronald L. Grudziecki | 24,970 | William C. Rowland | 30,888 | Harold R. Brown III | 36,341 |
| Frederick G. Michaud, Jr. | 26,003 | T. Gene Dillahunty | 25,423 | Allen R. Baum | 36,086 |
| Alan E. Kopecki | 25,813 | Patrick C. Keane | 32,858 | Steven M. duBois | 35,023 |
| Regis E. Slutter | 26,999 | B. Jefferson Boggs, Jr. | 32,344 | Brian P. O'Shaughnessy | 32,747 |
| Samuel C. Miller, III | 27,360 | William H. Benz | 25,952 | Kenneth B. Leffler | 36,075 |
| Robert G. Mukai | 28,531 | Peter K. Skiff | 31,917 | Fred W. Hathaway | 32,236 |
| George A. Hovanec, Jr. | 28,223 | Richard J. McGrath | 29,195 | | |
| James A. LaBarre | 28,632 | Matthew L. Schneider | 32,814 | | |
| E. Joseph Gess | 28,510 | Michael G. Savage | 32,596 | | |

21839

and: ______________________________.

Address all correspondence to: James A. LaBarre
BURNS, DOANE, SWECKER & MATHIS, L.L.P.
P.O. Box 1404
Alexandria, Virginia 22313-1404

21839

Address all telephone calls to: James A. LaBarre at (703) 836-6620.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

| COMBINED DECLARATION FOR PATENT APPLICATION AND POWER OF ATTORNEY (CONT'D) (Includes Reference to Provisional and International (PCT) Applications) | Attorney's Docket No. 032326-169 |
|---|---|

1-00

| FULL NAME OF SOLE OR FIRST INVENTOR<br>CORON Jean-Sébastien | SIGNATURE | DATE<br>9/10/2007 |
|---|---|---|
| RESIDENCE (CITY & STATE/COUNTRY)<br>4 rue Léon Delagrange – 75015 PARIS - FRANCE FRX | | CITIZENSHIP<br>FRANCE |
| POST OFFICE ADDRESS (HOME ADDRESS)<br>4 rue Léon Delagrange – 75015 PARIS - FRANCE | | |
| FULL NAME OF SECOND JOINT INVENTOR, IF ANY | SIGNATURE | DATE |
| RESIDENCE (CITY & STATE/COUNTRY) | | CITIZENSHIP |
| POST OFFICE ADDRESS (HOME ADDRESS | | |
| FULL NAME OF THIRD JOINT INVENTOR, IF ANY | SIGNATURE | DATE |
| RESIDENCE (CITY & STATE/COUNTRY) | | CITIZENSHIP |
| POST OFFICE ADDRESS (HOME ADDRESS | | |
| FULL NAME OF FOURTH JOINT INVENTOR, IF ANY | SIGNATURE | DATE |
| RESIDENCE (CITY & STATE/COUNTRY) | | CITIZENSHIP |
| POST OFFICE ADDRESS (HOME ADDRESS | | |
| FULL NAME OF FIFTH JOINT INVENTOR, IF ANY | SIGNATURE | DATE |
| RESIDENCE (CITY & STATE/COUNTRY) | | CITIZENSHIP |
| POST OFFICE ADDRESS (HOME ADDRESS | | |
| FULL NAME OF SIXTH JOINT INVENTOR, IF ANY | SIGNATURE | DATE |
| RESIDENCE (CITY & STATE/COUNTRY) | | CITIZENSHIP |
| POST OFFICE ADDRESS (HOME ADDRESS | | |
| FULL NAME OF SEVENTH JOINT INVENTOR, IF ANY | SIGNATURE | DATE |
| RESIDENCE (CITY & STATE/COUNTRY) | | CITIZENSHIP |
| POST OFFICE ADDRESS (HOME ADDRESS | | |
| FULL NAME OF EIGHTH JOINT INVENTOR, IF ANY | SIGNATURE | DATE |
| RESIDENCE (CITY & STATE/COUNTRY) | | CITIZENSHIP |
| POST OFFICE ADDRESS (HOME ADDRESS | | |
| FULL NAME OF NINTH JOINT INVENTOR, IF ANY | SIGNATURE | DATE |
| RESIDENCE (CITY & STATE/COUNTRY) | | CITIZENSHIP |
| POST OFFICE ADDRESS (HOME ADDRESS) | | |
| FULL NAME OF TENTH JOINT INVENTOR, IF ANY | SIGNATURE | DATE |
| RESIDENCE (CITY & STATE/COUNTRY) | | CITIZENSHIP |
| POST OFFICE ADDRESS (HOME ADDRESS | | |